

**Listing of Claims:**

1. (Currently Amended) A method of incrementally rendering content in a content framework, comprising ~~steps of~~:
  - receiving a request for a portal page, wherein one or more portlets provide content for the portal page;
  - immediately returning a response message containing a first document, the first document representing results from portlets which have acquired their content; and
  - programmatically generating a mechanism for delivering an updated document if the first document does not represent results of all portlets.
2. (Original) The method according to Claim 1, wherein the programmatically generated mechanism comprises inclusion of a refresh trigger in the response message.
3. (Original) The method according to Claim 2, wherein the refresh trigger is a refresh header of the response message.
4. (Original) The method according to Claim 2, wherein the refresh trigger is encoded using syntax of a markup language.
5. (Original) The method according to Claim 4, wherein the markup language is HTML ("Hypertext Markup Language").
6. (Original) The method according to Claim 5, wherein the syntax comprises a "META" tag using an "HTTP-EQUIV" attribute syntax.
7. (Original) The method according to Claim 4, wherein the markup language is WML ("Wireless Markup Language").

8. (Original) The method according to Claim 4, wherein the markup language is i-mode format.

9. (Original) The method according to Claim 4, wherein the markup language is HDML ("Handheld Device Markup Language").

10. (Original) The method according to Claim 2, wherein a value on the refresh trigger specifies a time before which the programmatically generated mechanism does not execute.

11. (Original) The method according to Claim 2, wherein a value on the refresh trigger is computed as a time after which a sender of the portal page request automatically invokes the delivery of the updated document.

12. (Original) The method according to Claim 2, wherein a value on the refresh trigger is computed as a latest predicted completion time of a final one of the portlets which have not yet acquired their content.

13. (Original) The method according to Claim 12, wherein the value is determined by weighting actual fetch times of the portlets which have not yet acquired their content.

14. (Original) The method according to Claim 12, wherein the value is determined by adding a constant value to a largest of weighted actual fetch times of the portlets which have not yet acquired their content.

15. (Currently Amended) The method according to Claim 2, further comprising steps of:

receiving the response message by a client from which the request for the portal page was sent;

rendering, by the client, the first document from the received response message; and  
automatically sending a subsequent request for the portal page after waiting for a time  
specified by a value of the refresh trigger.

16. (Currently Amended) The method according to Claim 2, further comprising ~~steps~~  
~~of~~:

receiving a subsequent request for the portal page, the subsequent request having been  
automatically sent responsive to receiving the refresh trigger; and

returning a subsequent response comprising the updated document, responsive to  
receiving the subsequent request, the updated document being a subsequent version of the first  
document and representing results from portlets which have acquired their content thus far and  
which omits the refresh trigger only if all portlets have now acquired their content.

17. (Original) The method according to Claim 1, wherein the programmatically  
generated mechanism comprises creating a multipart document, and wherein the first document  
is embedded in a first of the parts of the multipart document.

18. (Original) The method according to Claim 17, wherein the first of the parts is  
preceded by a boundary string used to delimit parts of the multipart document.

19. (Original) The method according to Claim 18, wherein the first of the parts is  
followed by a terminating boundary string if the first document represents results from all  
portlets.

20. (Currently Amended) The method according to Claim 17, further comprising  
~~steps of~~:

receiving the response message by a client from which the request for the portal page was  
sent;

rendering, by the client, the first document from the first of the parts of the multipart  
document;

receiving, by the client, subsequent parts of the multipart document, each of the subsequent parts comprising a revised version of the first document; and  
rendering, by the client, the subsequent parts of the multipart document.

21. (Currently Amended) The method according to Claim 17, further comprising ~~steps of:~~

detecting that one or more of the portlets which had not acquired their content when the first document was returned in the response message have now acquired their content; and

sending, responsive to ~~the detecting step~~, a subsequent response message containing a revised version of the first document, the revised version representing results from the one or more portlets and being embedded in a subsequent part of the multipart document.

22. (Original) The method according to Claim 21, wherein the subsequent part is preceded by a boundary string used to delimit parts of the multipart document and is followed by a terminating boundary string if the revised version represents results from all portlets.

23. (Original) The method according to Claim 1, wherein the programmatically generated mechanism comprises programmatically inserting a hyperlink into the first document, wherein the inserted hyperlink can be used to explicitly request delivery of the updated document.

24. (Currently Amended) A method of incrementally rendering content in a content framework, comprising ~~steps of:~~

receiving a request for a portal page, wherein one or more portlets provide content for the portal page;

immediately returning a response message containing a first document, the first document representing results from portlets which have acquired their content; and

automatically delivering an updated document if the first document does not represent results of all portlets.

25. (Currently Amended) A method of incrementally rendering content in a content framework, comprising ~~steps of~~:
- receiving a request for a portal page frame, wherein one or more portlets provide content for the portal page frame;
  - immediately returning a response message containing a first mini-document, the first document representing results from portlets which have acquired their content; and
  - programmatically generating a mechanism for delivering an updated mini-document if the first mini-document does not represent results of all portlets.
26. (Original) The method according to Claim 25, wherein the programmatically generated mechanism comprises inclusion of a refresh header in the response message.
27. (Original) The method according to Claim 25, wherein the programmatically generated mechanism comprises inclusion of an syntax element in the response header, wherein the syntax element is encoded using a markup language.
28. (Original) The method according to Claim 27, wherein the markup language is HTML ("Hypertext Markup Language") and the syntax element comprises a "META" tag using an "HTTP-EQUIV" attribute syntax.
29. (Original) The method according to Claim 26, wherein a value on the refresh header is computed as a time after which a sender of the portal page frame request automatically invokes the delivery of the updated mini-document.
30. (Original) The method according to Claim 25, wherein the programmatically generated mechanism comprises creating a multipart document, and wherein the first mini-document is embedded in a first of the parts of the multipart document.
31. (Original) The method according to Claim 30, wherein the first of the parts is preceded by a boundary string used to delimit parts of the multipart document, and is followed

by a terminating boundary string if the first mini-document represents results from all portlets.

32. (Original) A system for incrementally rendering content in a content framework, comprising:

means for receiving a request for a portal page, wherein one or more portlets provide content for the portal page;

means for immediately returning a response message containing a first document, the first document representing results from portlets which have acquired their content; and

means for programmatically generating a mechanism for delivering an updated document if the first document does not represent results of all portlets.

33. (Original) The system according to Claim 32, wherein the programmatically generated mechanism comprises inclusion of a refresh trigger in the response message.

34. (Original) The system according to Claim 32, further comprising:  
means for receiving the response message by a client from which the request for the portal page was sent;

means for rendering, by the client, the first document from the received response message; and

means for automatically sending a subsequent request for the portal page after waiting for a time specified by a value of the refresh trigger.

35. (Original) The system according to Claim 32, further comprising:  
means for receiving a subsequent request for the portal page, the subsequent request having been automatically sent responsive to receiving the refresh trigger; and  
means for returning a subsequent response comprising the updated document, responsive to receiving the subsequent request, the updated document being a subsequent version of the first document and representing results from portlets which have acquired their content thus far and which omits the refresh trigger only if all portlets have now acquired their content.

36. (Original) The system according to Claim 32, wherein the programmatically generated mechanism comprises creating a multipart document, and wherein the first document is embedded in a first of the parts of the multipart document.

37. (Original) The system according to Claim 36, wherein the first of the parts is preceded by a boundary string used to delimit parts of the multipart document, and is followed by a terminating boundary string only if the first document represents results from all portlets.

38. (Original) The system according to Claim 36, further comprising:  
means for receiving the response message by a client from which the request for the portal page was sent;  
means for rendering, by the client, the first document from the first of the parts of the multipart document;  
means for receiving, by the client, subsequent parts of the multipart document, each of the subsequent parts comprising a revised version of the first document; and  
means for rendering, by the client, the subsequent parts of the multipart document.

39. (Original) The system according to Claim 36, further comprising:  
means for detecting that one or more of the portlets which had not acquired their content when the first document was returned in the response message have now acquired their content;  
and  
means for sending, responsive to the means for detecting, a subsequent response message containing a revised version of the first document, the revised version representing results from the one or more portlets and being embedded in a subsequent part of the multipart document.

40. (Original) The system according to Claim 39, wherein the subsequent part is preceded by a boundary string used to delimit parts of the multipart document and is followed by a terminating boundary string if the revised version represents results from all portlets.

41. (Original) The system according to Claim 32, wherein the programmatically generated mechanism comprises programmatically inserting a hyperlink into the first document, wherein the inserted hyperlink can be used to explicitly request delivery of the updated document.

42. (Original) A system for incrementally rendering content in a content framework, comprising:

means for receiving a request for a portal page frame, wherein one or more portlets provide content for the portal page frame;

means for immediately returning a response message containing a first mini-document, the first document representing results from portlets which have acquired their content; and

means for programmatically generating a mechanism for delivering an updated mini-document if the first mini-document does not represent results of all portlets.

43. (Original) The system according to Claim 42, wherein the programmatically generated mechanism comprises inclusion of a refresh header in the response message.

44. (Currently Amended) The ~~method~~ system according to Claim 42, wherein the programmatically generated mechanism comprises inclusion of an syntax element in the response header, wherein the syntax element is encoded using a markup language.

45. (Original) The system according to Claim 42, wherein the programmatically generated mechanism comprises creating a multipart document, and wherein the first mini-document is embedded in a first of the parts of the multipart document.

46. (Currently Amended) A computer program product incrementally rendering content in a content framework, the computer program product embodied on one or more computer-usable media and comprising:

computer readable program code ~~means for receiving~~ configured to receive a request for



a portal page, wherein one or more portlets provide content for the portal page;

computer readable program code ~~means for immediately returning~~ configured to immediately return a response message containing a first document, the first document representing results from portlets which have acquired their content; and

computer readable program code ~~means for programmatically generating~~ configured to programmatically generate a mechanism for delivering an updated document if the first document does not represent results of all portlets.

47. (Original) The computer program product according to Claim 46, wherein the programmatically generated mechanism comprises inclusion of a refresh trigger in the response message.

48. (Currently Amended) The computer program product according to Claim 46, further comprising:

computer readable program code ~~means for receiving~~ configured to receive the response message by a client from which the request for the portal page was sent;

computer readable program code ~~means for rendering~~ configured to render, by the client, the first document from the received response message; and

computer readable program code ~~means for automatically sending~~ configured to automatically send a subsequent request for the portal page after waiting for a time specified by a value of the refresh trigger.

49. (Currently Amended) The computer program product according to Claim 46, further comprising:

computer readable program code ~~means for receiving~~ configured to receive a subsequent request for the portal page, the subsequent request having been automatically sent responsive to receiving the refresh trigger; and

computer readable program code ~~means for returning~~ configured to return a subsequent response comprising the updated document, responsive to receiving the subsequent request, the updated document being a subsequent version of the first document and representing results from

portlets which have acquired their content thus far and which omits the refresh trigger only if all portlets have now acquired their content.

50. (Original) The computer program product according to Claim 46, wherein:  
the programmatically generated mechanism comprises creating a multipart document;  
the first document is embedded in a first of the parts of the multipart document;  
the first of the parts is preceded by a boundary string used to delimit parts of the multipart document; and  
the first of the parts is followed by a terminating boundary string only if the first document represents results from all portlets.

51. (Currently Amended) The computer program product according to Claim 50, further comprising:

computer readable program code ~~means for receiving~~ configured to receive the response message by a client from which the request for the portal page was sent;

computer readable program code ~~means for rendering~~ configured to render, by the client, the first document from the first of the parts of the multipart document;

computer readable program code ~~means for receiving~~ configured to receive, by the client, subsequent parts of the multipart document, each of the subsequent parts comprising a revised version of the first document; and

computer readable program code ~~means for rendering~~ configured to render, by the client, the subsequent parts of the multipart document.

52. (Currently Amended) The computer program product according to Claim 50, further comprising:

computer readable program code ~~means for detecting~~ configured to detect that one or more of the portlets which had not acquired their content when the first document was returned in the response message have now acquired their content; and

computer readable program code ~~means for sending~~ configured to send, responsive to the computer readable program code ~~means for detecting~~ configured to detect, a subsequent response

message containing a revised version of the first document, the revised version representing results from the one or more portlets and being embedded in a subsequent part of the multipart document.

53. (Original) The computer program product according to Claim 52, wherein the subsequent part is preceded by a boundary string used to delimit parts of the multipart document and is followed by a terminating boundary string only if the revised version represents results from all portlets.